Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 31 (2019) 264–269

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

9th Conference on Learning Factories 2019

Development of a low cost machine vision based quality control system for a learning factory

Louis Louw[a,*], Marli Droomer[a]

[a]Industrial Engineering Department, Stellenbosch University, Joubert Street, Stellenbosch, 7600, South Africa

Abstract

Learning Factories provide a promising environment for developing the competencies required from a future workforce to apply and integrate technologies associated with digitalised production environments and cyber-physical systems. This paper describes a student project for the development and implementation of a low cost machine vision based quality control system within a Learning Factory. A prototype system was developed using low cost hardware and open source software freely available. The system will be used towards further research and development of more intelligent manufacturing systems within the Learning Factory, based on machine vision. A second benefit was student competency development through self-learning and experimentation. It serves to illustrate how the education as well as research goals of a Learning Factory can be addressed simultaneously through student projects.

Keywords: Machine vision; Quality control; Learning Factory; Digitilisation

1. Introduction

A Learning Factory can be defined as a "dedicated facility that mimics real production processes and environments and is used to develop competencies of present and future industry personnel" [1]. Tisch et al [2] developed a morphology for structuring and classifying Learning Factories. They identify three high level goals of learning factories: education, vocational training, and research. Through the practical application of visually and physically interacting with the manufacturing system, the student will experience better knowledge transfer and retention [3]. The method of action-based learning has become universally referred to as 'learning by doing'.

* Corresponding author. Tel.: +27 21 808 4448.
E-mail address: louisl@sun.ac.za

2351-9789 © 2019 The Authors. Published by Elsevier B.V.
Peer review under the responsibility of the scientific committee of the 9th Conference on Learning Factories.
10.1016/j.promfg.2019.03.042

Learning Factories can be structured so that learners apply theory and calculation analysis to predefined problems or so that the learner employs heuristic methods to iteratively provide suggestions for problems which are not previously defined [1].

Associated with the 4th industrial revolution is a new world of digitalised, decentralized, autonomous real-time production which has brought about a shift of skills required in the industrial production setting [4,5]. Sacky and Bester [6] conducted a study to identify curricula enhancements requirement to accommodate Industrie 4.0 skills in Industrial Engineering higher education programmes. They identified the following areas needing attention in Industrial Engineering curriculums: data science and advanced analytics; advanced simulation and virtual plant modelling; data communication and networks and system automation; novel human-machine interfaces; digital-to-physical transfer technologies such as 3D printing; closed-loop integrated product and process quality control/ management systems; real-time inventory and logistics optimization systems; and teaching and learning demonstration infrastructure.

Learning Factories provide a promising environment for developing the competencies required from a future workforce to apply and integrate such technologies associated with Industry 4.0 [4,7]. It allows prototyping and experimentation with technology, which builds competency in the use and application of the technology. This also builds innovation competencies in young engineers, with skills such as problem solving, creativity and systems thinking, as well as the ability to act self-organized in unknown situations. This paper describes such an innovation project in which the student was tasked with the development of a machine vision based quality control prototype. The student had no prior teaching or theory on machine vision based systems, and had to define the problem, the approach as well as the solution by herself. The student was self-reliant on sourcing relevant information and technologies, and therefor had to make use of open source software systems and low cost hardware. This project is one of a range of similar project types (but with different focus areas) conducted in the Learning Factory environment, and serves to illustrate how the education as well as research goals of a Learning Factory can be addressed simultaneously. In the following sections, the problem context and approach is first described, followed by a description of the project execution from theory to requirements, to design and prototyping. The paper concludes with some learnings and observations.

2. Problem Context

As mentioned in the previous section, a key goal of Learning Factories is effective competency development. The higher level competencies identified for this project-based learning in the SLF included the following: problem solving, project management, prototyping, creativity, systems thinking and design. Lower level competencies included software and database design, software integration with hardware, as well as digitalisation.

Machine vision is an encompassing term which includes the integration of both hardware and software for practical application. Sensing devices are key for the implementation of Internet-of-things, and machine or computer vision can very efficiently and effectively be applied as a sensing device within a smart production environment [9]. Machine vision technology is therefore a good use case for skills development towards digitalision, automation, and cyber-physical systems. The Stellenbosch Learning Factory (SLF) was initiated in 2015 as a place where students can experience action-based learning. The initial focus of the Learning Factory was to teach students production system design and management methods. The SLF allows a student to follow the production and assembly process of model-trains which replicate the South-African Metrorail's trains on a scaled-down version. It is currently under internal construction to allow the facility to be fully utilised in the education of students, as well conducting research and development projects. The products assembled in the SLF are two variants of a model train: (1) a two-door driver trains and (2) a three-door cabin train, as seen in Fig. 1 (a)**Fehler! Verweisquelle konnte nicht gefunden werden.**. An analysis was conducted in the SLF to identify potential for application of machine vision technology. Quality management was identified as an ideal application for the technology [10,11]. The current quality inspection in the SLF is a manual final inspection (refer to Fig. 1 (b) for positioning within assembly). The quality inspection worker checks if all screws are fixed properly and check if all components are correctly positioned. This manual quality check is very inefficient and inconsistent. Workers loose concentration and make mistakes, and sometimes forget to add the quality information to the information system, or enter incorrect information.

Automation of this process can therefore help to increase productivity, as well as the quality of the product shipped and the quality of the production system information.

a b

Fig. 1. (a) Driver train (top), cabin train (bottom); (b) assembly layout indicating quality inspection station.

3. Approach

Understanding and being able to apply the engineering design process is a critical element of engineering education. Atman et al [12] conducted a study comparing the engineering design processes of students and expert practitioners. Their study makes the argument that problem scoping and information gathering are important competencies for engineering students to develop.

Various types of models and processes exist in the literature to describe the engineering design and development process [13]. Wynn et al [13] describes four main groups of procedural models that provide prescriptive guidelines for the design and problem-solving activities. The *first group* of models are on a conceptual level and describe overall recommended design strategies. Most of these strategies include the three main steps of analysis, synthesis and evaluation. The *second group* of models are more systematic and support the execution of design steps. These include approaches such as the Pahl and Beitz model, Axiomatic Design by Suh, Triz, etc [13]. The *third set of models* are more focused om providing procedures for solving problems encountered during the design and development process, such as the Plan-Do-Check-Act (PCDA) cycle, Define-Model-Analyse-Improve-Control (DMAIC), A3 problem solving, etc. The *fourth group* of models focus more on describing the interaction processes between different stakeholders in the design process. For this project, an engineering design process based on the first group of procedural models was followed, but with a problem description directing the solution approach towards a machine vision based solution in order to address the competency requirements for the project, as well as the need of the SLF for automation and digitalisation. The main steps included problem definition, background research, requirements specification, hardware and software selection, prototype development, testing and refinement, and communication of results.

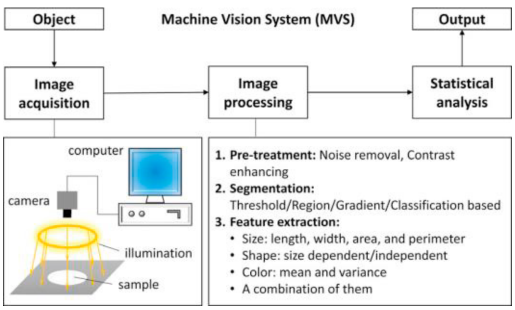


Fig. 2. Elements of a machine vision system [14].

4. Requirements and Software/ Hardware Selection

Various open source computer vision software exist. The following were considered for this project: OpenCV, SimpleCV, TensorFlow, and CVIPtools. Key selection criteria for the computer vision software included compatibility with the Linux operating system and Python, should have a great support network, be updated in the last few years, and should be open source. OpenCV was selected as it fulfills all these criteria.

A GUI software package that is compatible with Python and with MariaDB was required. Python's GUI package tkinter was used for the displaying of the relevant quality indicators.

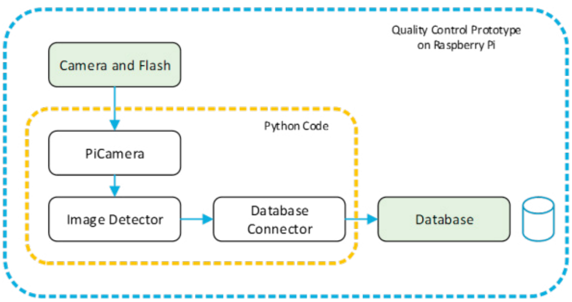
Requirements for the Relational Database Management System (RDMS) needed for this project included that the software must be open source (able to operate on Linux), be able to execute Structure Query Language (SQL) statements, as well as be able to interpret Python triggers to manipulate or add data into the database. PostgreSQL, MySQL, MariaDB, and SQLite were considered. From these options, MariaDB was selected as it was easily available, easy to use, has better performance compared to MySQL and is very popular amongst open source users.

The main hardware components included a microcontroller and a camera module connected to the microcontroller. The primary criteria for the microcontroller were: the ability to connect a camera module, an easy to use interface, can run an open source operating system (Linux), has a Quad-Core processor, is WIFI-enabled, at the lowest possible cost. Raspberry Pi 3 Model B, Banana Pi M64, Beagleboard X15, as well as Odroid-C2 were considered. Since the Beagleboard and Odroid had no integrated Wi-Fi capabilities, these two were excluded. The Banana Pi M64 offers the same specifications as the Raspberry Pi, and meets the specified criteria, but is more expensive. The Raspberry Pi 3 was therefore selected. It has become a well-known product and has developed a big community of users creating and improving open source code that is available widely available and makes it easier to use the product. The main selection criteria for the camera module included compatibility with the Raspberry Pi 3, availability, and cost. The RPi Camera (D) was selected based on these three criteria. This module connects to the Raspberry Pi 3B with a 15-pin flexible ribbon to the CSI camera connector on the Raspberry Pi 3B.

5. Prototype development and testing

Before development, an integrated system design combining hardware and software was done. A complete structure of the hardware components and how they interact with each other and with the software packages can be seen in Fig. 3 (a). A camera and a camera flash connect through a PiCamera package. This package enables communication between Python and the Raspberry Pi camera. Image detection is done by using OpenCV. The results of the detection and image processing is stored in a database, that can be hosted on the Raspberry Pi or using cloud storage via wi-fi connection (using a database connector in Python). This is the backbone of the quality control prototype. A jig was designed and built to manually position the train directly underneath the camera, to reduce variability in the image processing phase (refer to Fig. 3 (b)). A GUI was designed consisting of a video frame that displays a live video feed, five buttons and a textbox. This allows for manual entry of the Chassis ID after placing the train in the jig. The software will check if the ID matches the one in the database, and the image recognition procedure will check if the train is correctly placed with the correct view. Notifications are displayed to the worker.

a



b

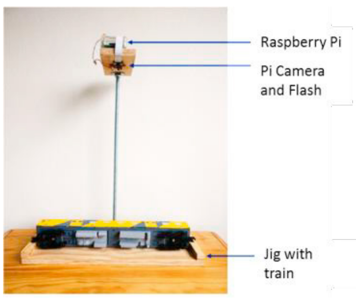


Fig. 3. (a) Quality control prototype component setup, (b) actual setup on jig.

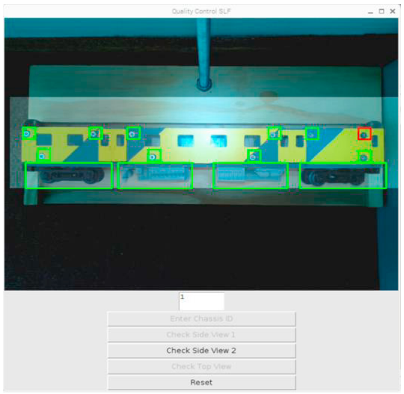


Fig. 4. Machine vision GUI and output indicating defects with red squares.

Images are captured for each view of the train (refer to Fig. 4 for a side view image) on manual press of a button on the GUI (a button for each view of the train) and the developed software will check for any defects based on the following quality indicators: (1) presence of all screws on the side views, (2) correct number of seats detected, (3) correct model of the train, and (4) presence of all four components that are mounted to the bottom of the trains. Screws or correctly positioned parts are indicated by green squares, and missing screws/ parts are indicated by red squares. The screws are located at different angles from the camera and has variation in the lighting that falls on them. This implies that image interpretation techniques can be used to classify the screws. A classifier was trained with images of screws and holes by using the Logistic Regression, Naïve Bayes and the Random Forest algorithm techniques. These techniques were evaluated to see which classifier must be used for the prototype. The results can be seen in Fig. 5. Test Set 1 (Fig. 5a) was taken in the same environment as that of the training data, whereas Test Set 2 (Fig. 5b) was taken in a different environment. From Fig. 5 it can be seen that the Random Forest Classifier and the Logistic Regression classifier strive to 100% as the training set increases (test set 1 in the same environment as the training data). Since the prototype will only be used in the SLF, increasing the camera resolution is advantageous. This however comes at a cost of losing execution speed. It takes longer to execute because the number of pixels per object increases. From these results the Logistic Regression Classifier was chosen. This classifier performs the best on average and performs the best in unknown circumstances.

If the user identifies a faulty output from the system (e.g. a hole indicated as a screw or a screw indicated as a hole) the user can click on the image and the defect will be added to the classifier, for continuous improvement and training of the classifier. To identify the various types of trains, template matching was used. Four templates were created, a three-door template, a two-door template that indicates the back of the train, one that indicates the front of the train, and a top view template. The developed machine vision system does template matching and indicates if it is a three-door or a two-door train.

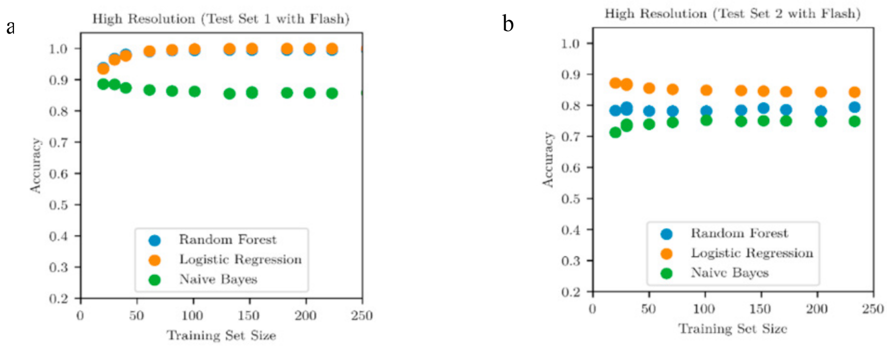


Fig. 5. Classifier accuracy for training done: (a) in same environment as test set; (b) different environments.

6. Conclusion

The machine vision based quality control prototype successfully identified the train, missing screws on the side views, as well as the presence and the location of the bottom parts of the train. The system was tested in the Learning Factory and found that initially it does not have 100% success rate (due to small training data sets). A continuous improvement strategy was therefore designed for the prototype to continuously train the classifiers ensuring it gets better with time. Future potential improvements include the use of RFID scanners to automatically log the train's Chassis ID in the quality system, as well as the use of Business intelligence (BI) software to analyse data and identify possible trends, as well as visually display these analyses. The system can also be incorporated in a robotic arm with built-in camera module, in order to automate the quality inspection operation.

The system developed serve as a building block towards more intelligent manufacturing systems within the learning factory, based on machine vision. It could be developed at very low cost. The main cost components were the Raspberry Pi microcontroller and the Pi Camera module. The combined cost for these two components was about €65. The software were all open source and free to use. A second benefit from the project included skills development for the student. This included problem solving competencies following an engineering design process, research competencies, as well as technical skills related to machine vision systems. It therefore demonstrated how learning factories can be used as a development and test environment while simultaneously developing competencies through an action-based learning approach.

References

[1] M. Tisch, C. Hertle, E. Abele, J. Metternich, R. Tenberg, Learning factory design: a competency-oriented approach integrating three design levels, International Journal of Computer Integrated Manufacturing. 29 (2016) 1355–1375.

[2] M. Tısch, F. Ranz, E. Abele, J. Metternıch, H. Vera, Learning factory morphology – Study of form and structure of an innovative learning approach in the manufacturing domain, Turkish Online Journal of Educational Technology. 2015 (2015) 356–363.

[3] B.J.S. Barron, D.L. Schwartz, N.J. Vye, A. Moore, A. Petrosino, L. Zech, J.D. Bransford, Doing With Understanding: Lessons From Research on Problem- and Project-Based Learning, Journal of the Learning Sciences. 7 (1998) 271–311.

[4] S. Erol, A. Jäger, P. Hold, K. Ott, W. Sihn, Tangible Industry 4.0: A Scenario-Based Approach to Learning for the Future of Production, Procedia CIRP. 54 (2016) 13–18.

[5] A. Richert, M. Shehadeh, F. Willicks, S. Jeschke, Digital Transformation of Engineering Education - Empirical Insights from Virtual Worlds and Human-Robot-Collaboration, International Journal of Engineering Pedagogy (IJEP). 6 (2016) 23.

[6] S.M. Sackey, A. Bester2, Industrial Engineering Curriculum in Industry 4.0 in a South African Context, The South African Journal of Industrial Engineering. 24 (2013) 175–189.

[7] E. Abele, G. Chryssolouris, W. Sihn, J. Metternich, H. ElMaraghy, G. Seliger, G. Sivard, W. ElMaraghy, V. Hummel, M. Tisch, S. Seifermann, Learning factories for future oriented research and education in manufacturing, CIRP Annals - Manufacturing Technology. 66 (2017) 803–826.

[8] V. Nandini, R. Deepak Vishal, C. Arun Prakash, S. Aishwarya, A Review on Applications of Machine Vision Systems in Industries, Indian Journal of Science and Technology. 9 (2016).

[9] J. Posada, C. Toro, I. Barandiaran, D. Oyarzun, D. Stricker, R. De Amicis, E.B. Pinto, P. Eisert, J. Döllner, I. Vallarino, Visual Computing as a Key Enabling Technology for Industrie 4.0 and Industrial Internet, IEEE Computer Graphics and Applications. 35 (2015) 26–40.

[10]H. Foidl, M. Felderer, Research challenges of industry 4.0 for quality management, in: Lecture Notes in Business Information Processing, 2016.

[11]S.S. A Sader, P. István Husti, M. Daróczi, SUGGESTED INDICATORS TO MEASURE THE IMPACT OF INDUSTRY 4.0 ON TOTAL QUALITY MANAGEMENT, n.d. https://stumejournals.com/journals/i4/2017/6/298/pdf (accessed November 9, 2018).

[12]C.J. Atman, R.S. Adams, M.E. Cardella, J. Turns, S. Mosborg, J. Saleem, Engineering design processes: A comparison of students and expert practitioners, in: Journal of Engineering Education, John Wiley & Sons, Ltd, 2007: pp. 359–379.

[13]D.C. Wynn, P.J. Clarkson, Process models in design and development, Research in Engineering Design. 29 (2018) 161–202.

[14]H. Hong, X. Yang, Z. You, F. Cheng, Visual quality detection of aquatic products using machine vision, Aquacultural Engineering. 63 (2014) 62–71.

[15]N. Perveen, D. Kumar, I. Bhardwaj, An Overview on Template Matching Methodologies and its Applications, 2 (2013) 988–995.